



MANUAL TECNICO PARA PAGINA WEB DE REPOSTERIA “DULECE ENCANTO”

GUZMAN CHOQUE MARCO ANTONIO

INDICE

Conceptos avanzados de JavaScript y su uso profesional	2
1. DOMContentLoaded.....	2
2. toggle	2
3. classList	2
4. textContent	2
5. contains	2
6. remove	3
7. scrollIntoView	3
8. behavior: "smooth"	3
9. resize	3
10. innerWidth	3
11. filter	4
12. target.....	4
13. appendChild.....	4
14. getElementsByName	4
15. querySelector	4
16. checked	5
17. stopPropagation	5
18. push.....	5
19. find.....	5
20. onclick	5
21. add.....	5
22. preventDefault	6
23. createElement.....	6
24. setTimeout.....	6
25. style	6
26. animation	6
27. keydown	7
28. getAttribute	7
29. preventDefault (recordatorio final).....	7

Conceptos avanzados de JavaScript y su uso profesional

A continuación te detallo cada término pero en un nivel **profesional/avanzado**, con enfoque en arquitectura, buenas prácticas y casos de uso reales en desarrollo web moderno.

1. DOMContentLoaded

Evento que indica que **el documento HTML ya fue completamente parseado**, pero **antes de que se carguen imágenes, estilos o recursos externos**.

Se usa para **inicializar scripts temprano**, evitando esperar a window.onload.

Aplicación avanzada:

- Inicializar componentes UI antes de cargas pesadas.
- Montar listeners globales o sistemas de routing en SPAs sin framework.
- Preprocesar nodos, copiar plantillas o renderizar contenido dinámico.

2. toggle

Método que alterna clases u otros valores booleanos.

Uso avanzado:

- Manejo de **máquinas de estados UI**.
- Encapsular lógica de apertura/cierre de componentes (menus, modales, tooltips).
- Evitar if/else redundantes en interacciones complejas.

3. classList

Permite manipular clases de un elemento sin manejar texto manualmente.

Uso avanzado:

- Gestión de clases según **reglas de estilo o temas dinámicos**.
- Implementar arquitecturas como BEM o utilidades JS sobre CSS.
- Activar sistemas de animación a partir de transiciones CSS controladas por JS.

4. textContent

Accede o modifica el contenido textual de un nodo.

Aplicación avanzada:

- Sanitización de texto para prevenir XSS cuando no se permite HTML.
- Actualización masiva de UI sin repintar HTML completo.
- Sistemas de internacionalización (i18n) con claves dinámicas.

5. contains

Revisa si un elemento contiene otro elemento o clase.

Uso avanzado:

- Validación estructural en árboles DOM complejos.
- Delegación de eventos (event delegation) en contenedores grandes.
- Prevención de cierres de menú cuando se hace clic dentro de él.

6. remove

Elimina nodos del DOM.

Uso avanzado:

- Limpieza de componentes dinámicos (SPA sin frameworks).
- Desmontaje de listeners y nodos para liberar memoria.
- Patrón disposal/cleanup para evitar memory leaks.

7. scrollToView

Desplaza la página hacia un elemento.

Uso avanzado:

- Navegación semántica automatizada en páginas largas.
- Sistemas de indexado, guías o tours interactivos.
- Personalización del comportamiento scroll en aplicaciones con contenedores.

8. behavior: "smooth"

Parte de configuraciones del scroll para animar desplazamientos.

Uso avanzado:

- Mejor UX en navegación de secciones.
- Interfaz más fluida en dashboards, paneles o formularios largos.
- Combinado con IntersectionObserver para crear efectos dinámicos.

9. resize

Evento del navegador al cambiar el tamaño de ventana.

Uso avanzado:

- Adaptar componentes en tiempo real sin recargar la página.
- Recalcular layouts dependientes del tamaño (grids dinámicos, sliders).
- Optimizar performance con debounce/throttle.

10. innerWidth

Ancho actual de la ventana.

Aplicación avanzada:

- Implementar breakpoints personalizados desde JS.

- Activar o desactivar funcionalidades según tamaño (feature toggles responsivos).
- Cálculos para canvas, gráficos o containers adaptables.

11. filter

Método avanzado de arrays para obtener subconjuntos.

Aplicación avanzada:

- Filtrado dinámico de catálogos, tablas o grids.
- Pipelines funcionales con múltiples transformaciones.
- Validaciones complejas basadas en condiciones encadenadas.

12. target

Elemento que originó un evento.

Uso avanzado:

- Delegación de eventos en listas dinámicas.
- Detección de clics en componentes hijos sin listeners duplicados.
- Sistemas de routing basados en atributos dinámicos.

13. appendChild

Agrega nodos al DOM.

Uso avanzado:

- Renderizado dinámico sin frameworks (templating nativo).
- Construcción incremental del DOM para optimizar performance.
- Integración de componentes creados por librerías externas.

14. getElementsByName

Selecciona elementos por atributo name.

Uso avanzado:

- Formularios complejos con múltiples inputs del mismo tipo.
- Manejo de grupos de radio/checkbox avanzados.
- Validación dinámica por categorías o secciones.

15. querySelector

Selecciona elementos usando selectores CSS.

Aplicación avanzada:

- Unificar lógicas de selección con CSS.
- Implementación de componentes UI altamente personalizados.
- Creación de utilidades de selección reutilizables.

16. checked

Propiedad booleana de checkboxes y radios.

Uso avanzado:

- Formularios reactivos sin frameworks.
- Lógica condicional en configuradores o filtros.
- Control de selección múltiple en interfaces complejas.

17. stopPropagation

Detiene el burbujeo de un evento.

Uso avanzado:

- Evitar conflictos entre listeners anidados.
- Controlar comportamientos en menús, dropdowns o modales.
- Implementar sistemas de lógica jerárquica de eventos.

18. push

Agrega elementos a un array.

Uso avanzado:

- Construcción incremental de datasets.
- Sistemas de logs y auditoría en memoria.
- Almacenamiento temporal de sesiones o historial.

19. find

Busca un elemento según una condición.

Uso avanzado:

- Búsquedas optimizadas sobre estructuras de negocio.
- Implementación de selección rápida en catálogos grandes.
- Localización de entidades únicas con lógica de negocio.

20. onclick

Asignación de un evento de clic.

Uso avanzado:

- Definir handlers globales en aplicaciones pequeñas.
- Extender comportamientos de componentes generados dinámicamente.
- Modificar comportamientos de terceros sin alterar su código fuente.

21. add

Agregar una clase, listener u otro valor.

Uso avanzado:

- Gestión granular de estilos.
- Aplicar temas, validaciones o estados UI.
- Activar animaciones controladas.

22. preventDefault

Previene la acción por defecto de un evento.

Uso avanzado:

- Formularios completamente controlados por JS.
- Implementación de drag & drop personalizado.
- Creación de shortcuts de teclado o navegación no estándar.

23. createElement

Crea un nodo HTML desde cero.

Uso avanzado:

- Motor de rendering manual sin frameworks.
- Generación de elementos según datos de APIs.
- Creación de plantillas y patrones UI reutilizables.

24. setTimeout

Ejecuta una función con retraso.

Uso avanzado:

- Control de animaciones basadas en tiempo.
- Retrasos en validaciones o requests.
- Timeouts de procesos asíncronos o simulación de carga.

25. style

Manipulación directa de estilos inline.

Uso avanzado:

- Personalización específica sin alterar CSS global.
- Cambios reactivos que dependen de valores dinámicos.
- Coordinación de animaciones complejas.

26. animation

Propiedad CSS controlada desde JS.

Uso avanzado:

- Coreografía de animaciones sincronizadas.

- Efectos de entrada/salida controlados por eventos.
- Sistemas avanzados de transición en single-page apps.

27. keydown

Evento de teclado.

Uso avanzado:

- Atajos avanzados (keyboard shortcuts).
- Control de videojuegos, editores, o apps interactivas.
- Accesibilidad: navegación mediante teclado.

28. getAttribute

Obtiene el valor de un atributo HTML.

Uso avanzado:

- Lectura de data-attributes para comportamientos dinámicos.
- Configuración de componentes UI mediante HTML declarativo.
- Routing basado en atributos personalizados.

29. preventDefault (recordatorio final)

Clave en:

- Formularios controlados.
- Navegación SPA.
- Interacciones customizadas como sliders, drag events y shortcuts.